Операционные системы

лекция 3

hexlet.org

Сегодня в лекции

- Как процессы представлены в системе и как ОС управляет ими
- Состояния процессов
- Структуры данных
- Как ОС использует эти структуры

ОС должна

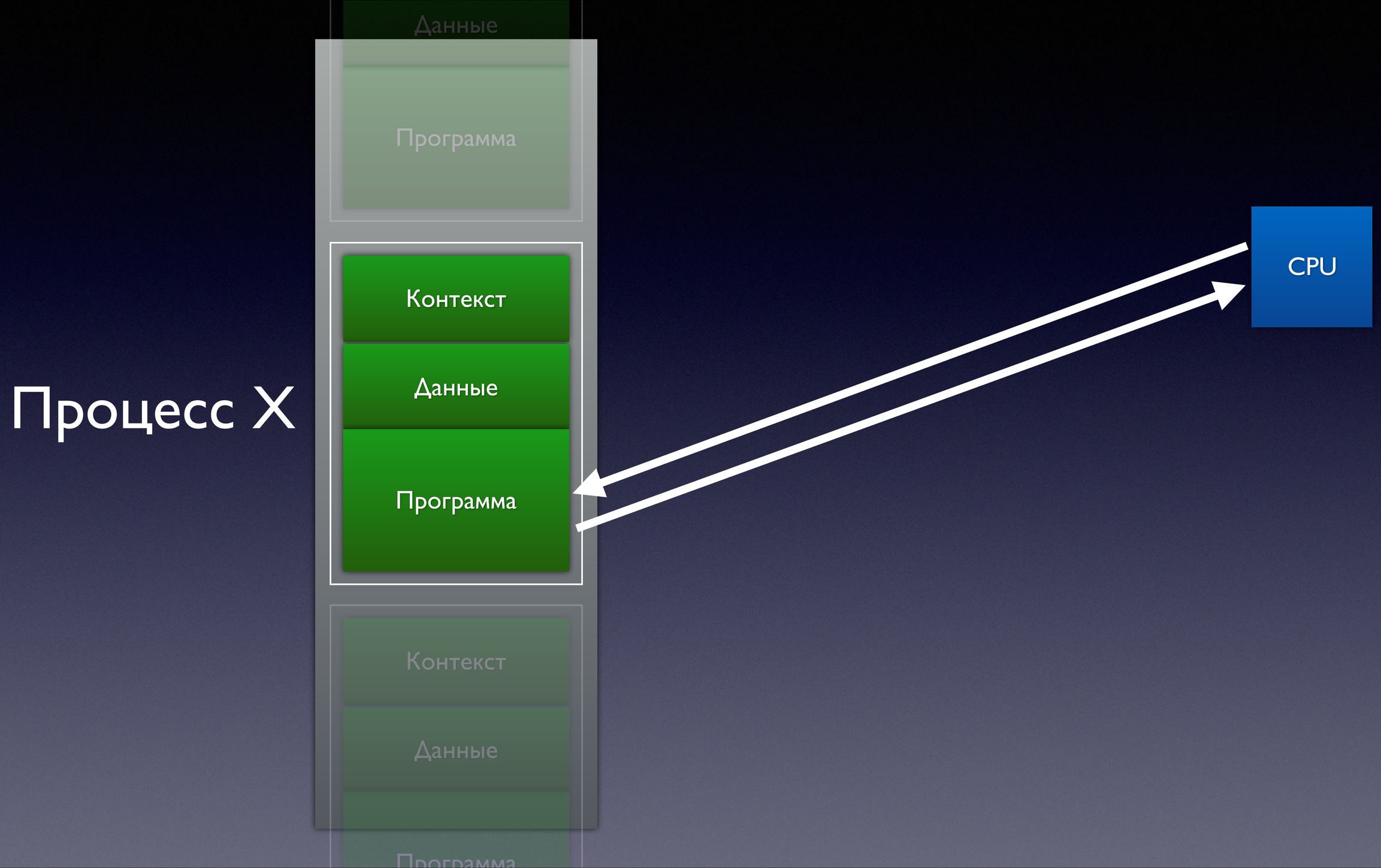
- Управлять запуском нескольких процессов
- Предоставлять необходимые ресурсы процессу и защищать ресурсы процесса от остальных
- Позволять процессам обмениваться информацией
- Позволять процессам синхронизироваться

Не забываем что

- Компьютеры “состоят” из нескольких типов ресурсов
- Компьютерные приложения выполняют конкретные задачи
- Невыгодно писать приложения для конкретной аппаратной платформы
- ОС предоставляет интерфейс для доступа к ресурсам

ОС управляет запуском приложений

- Ресурсы становятся доступными нескольким приложениям
- Процессор переключается между несколькими приложениями
- Процессор и устройства ввода/вывода могут и должны использоваться эффективно


Данные
Программа
Контекст
Данные
Программа
Контекст
Данные
Программа
Процесс X
CPU

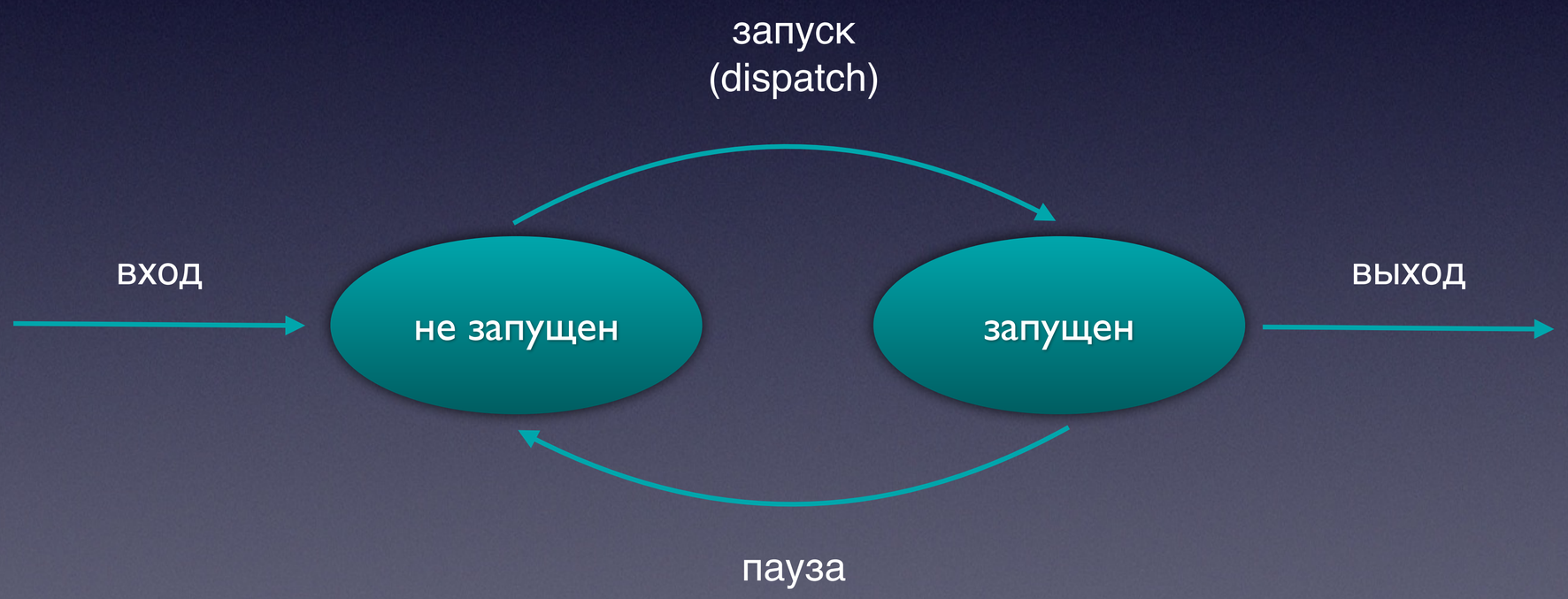
Процесс

- Идентификатор
- Состояние
- Приоритет
- Счетчик команд (Program counter)
- Указатели на память
- Контекст
- Информация о статусе I/O
- Другая информация

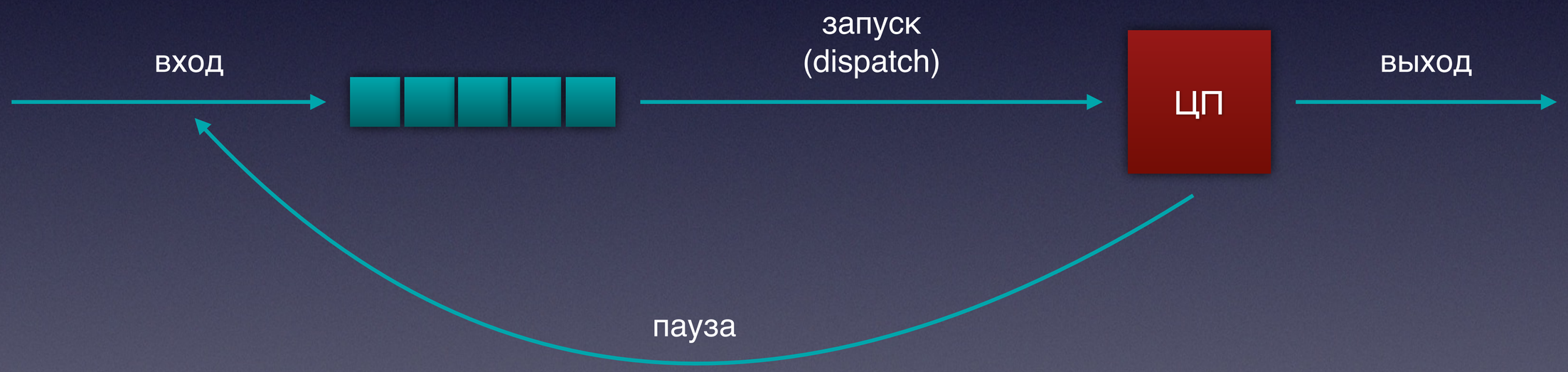
Process Control Block

- Структура данных, используемая ОС для представления процесса

Простая модель состояний процесса



Очередь



Жизнь и невероятные приключения процесса

- Создание (рождение)
- Уничтожение (смерть)

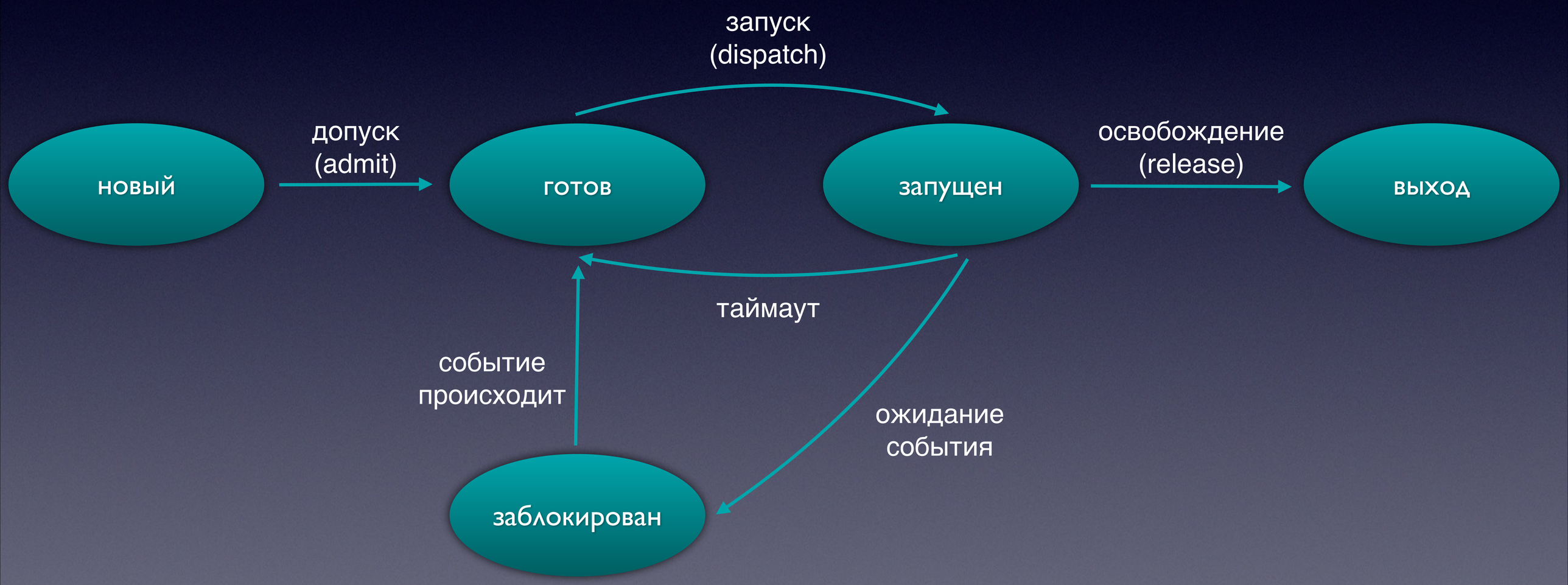
Создание

- ОС создает структуру данных для процесса
- Процессы обычно создаются самой ОС, но часто процессы могут создавать новые процессы
  - process spawning:
    - родительский процесс -> дочерний процесс

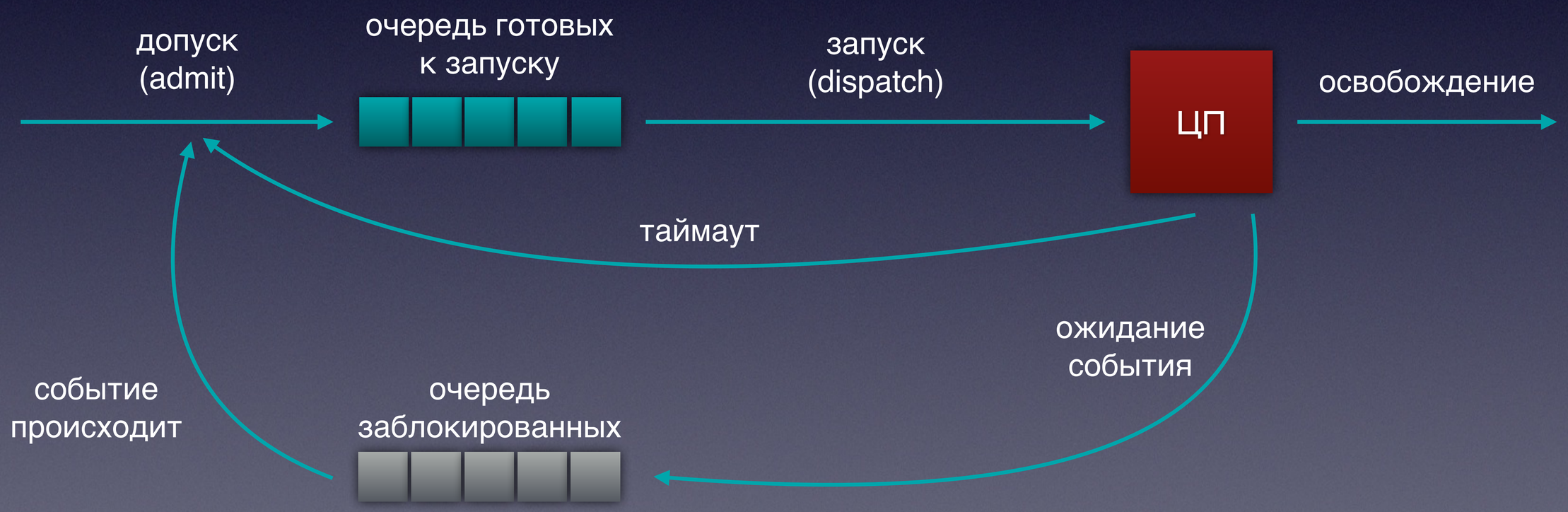
Уничтожение

- Необходимо чтобы было понятно, когда процесс завершен
  - Сигнал HALT
  - Действие пользователя
  - Ошибка
  - Завершение родительского процесса

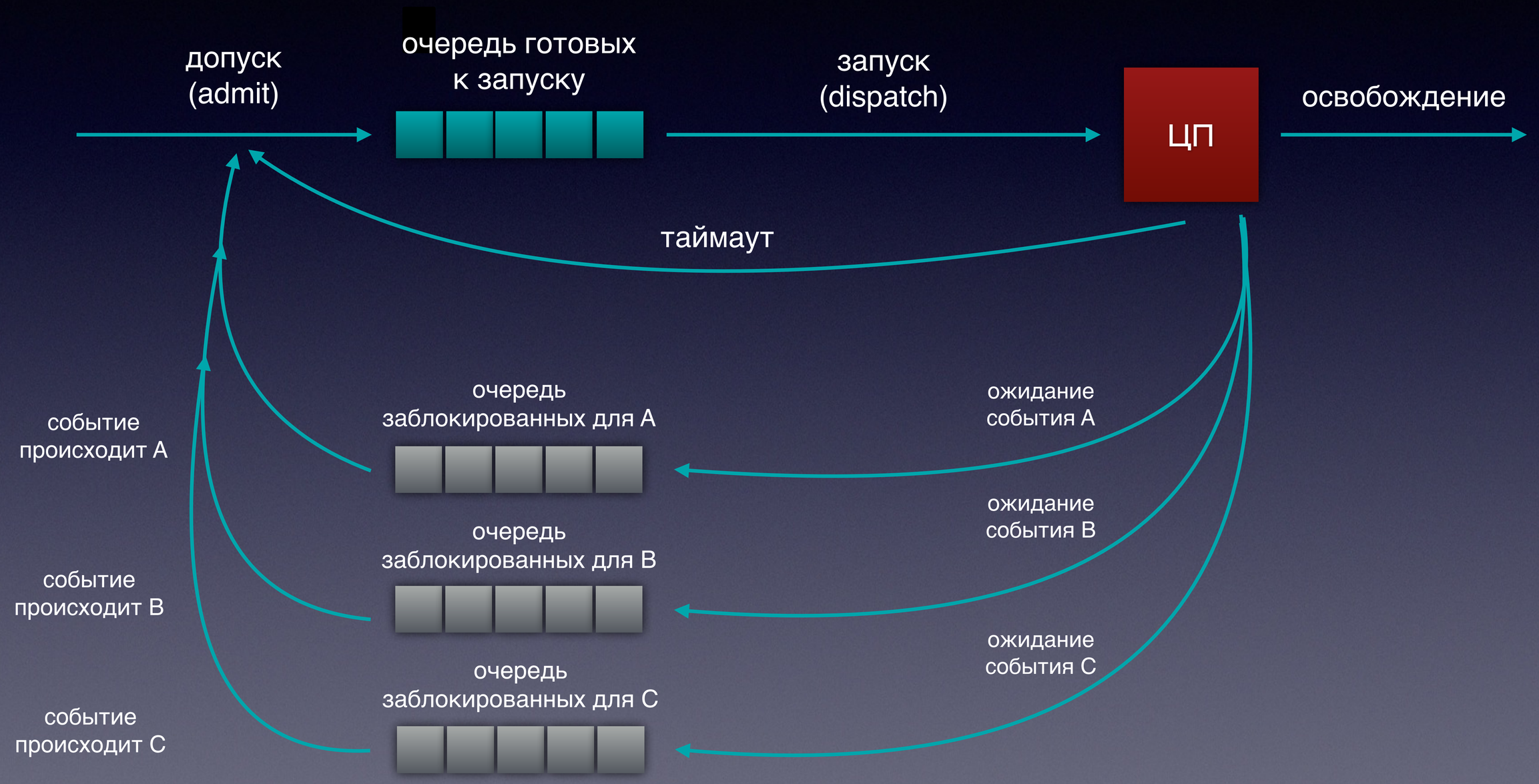
Пять состояний



Двойная очередь



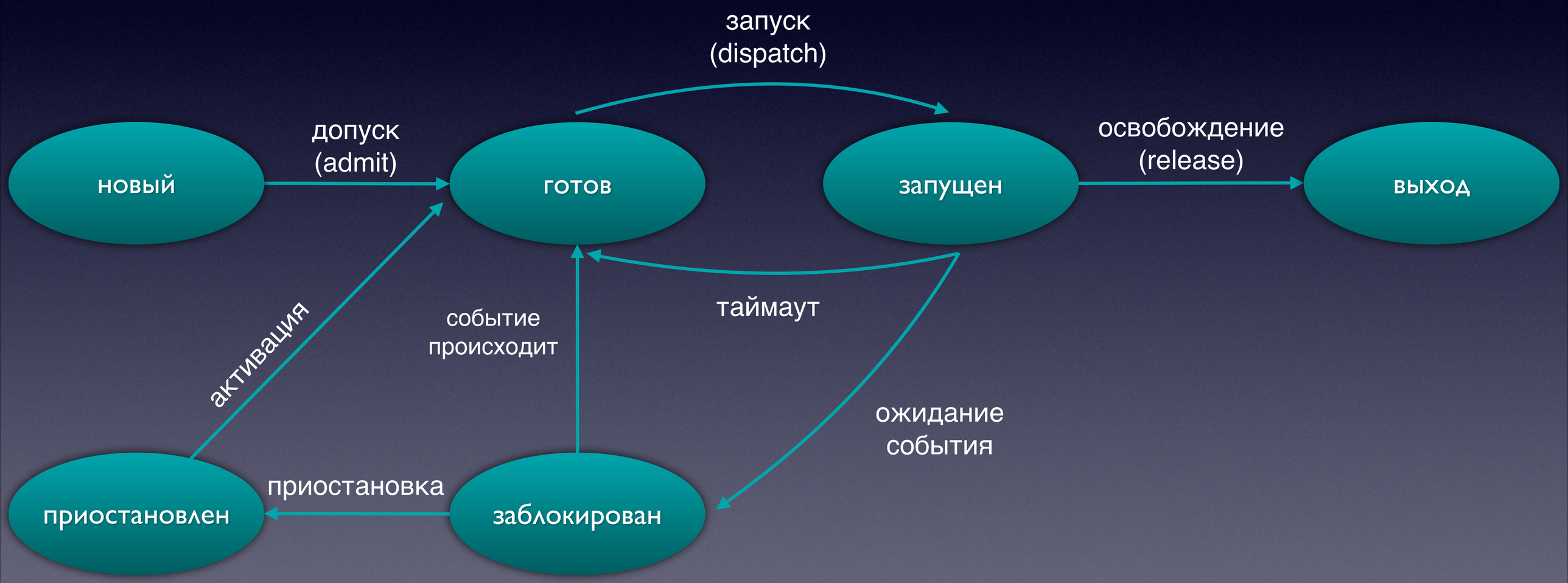
Множественная очередь



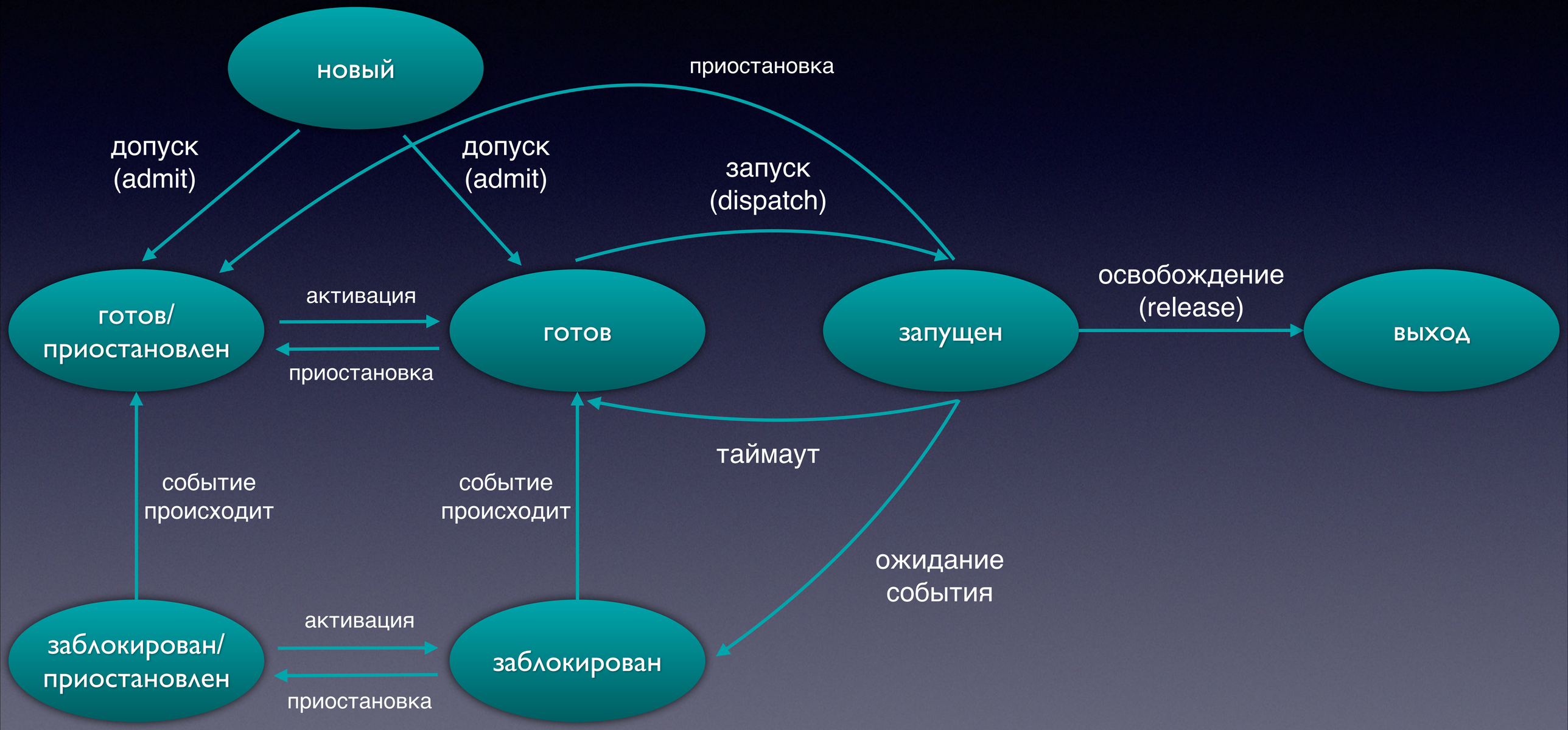
Приостановленные (suspended) процессы

- Возможно такое, что все процессы ждут I/O
- Их можно перенести на диск чтобы освободить память для более актуальных процессов
- Заблокированный процесс становится приостановленным
- Два новых состояния:
  - Заблокирован/приостановлен
  - Готов/приостановлен

Одно приостановленное состояние



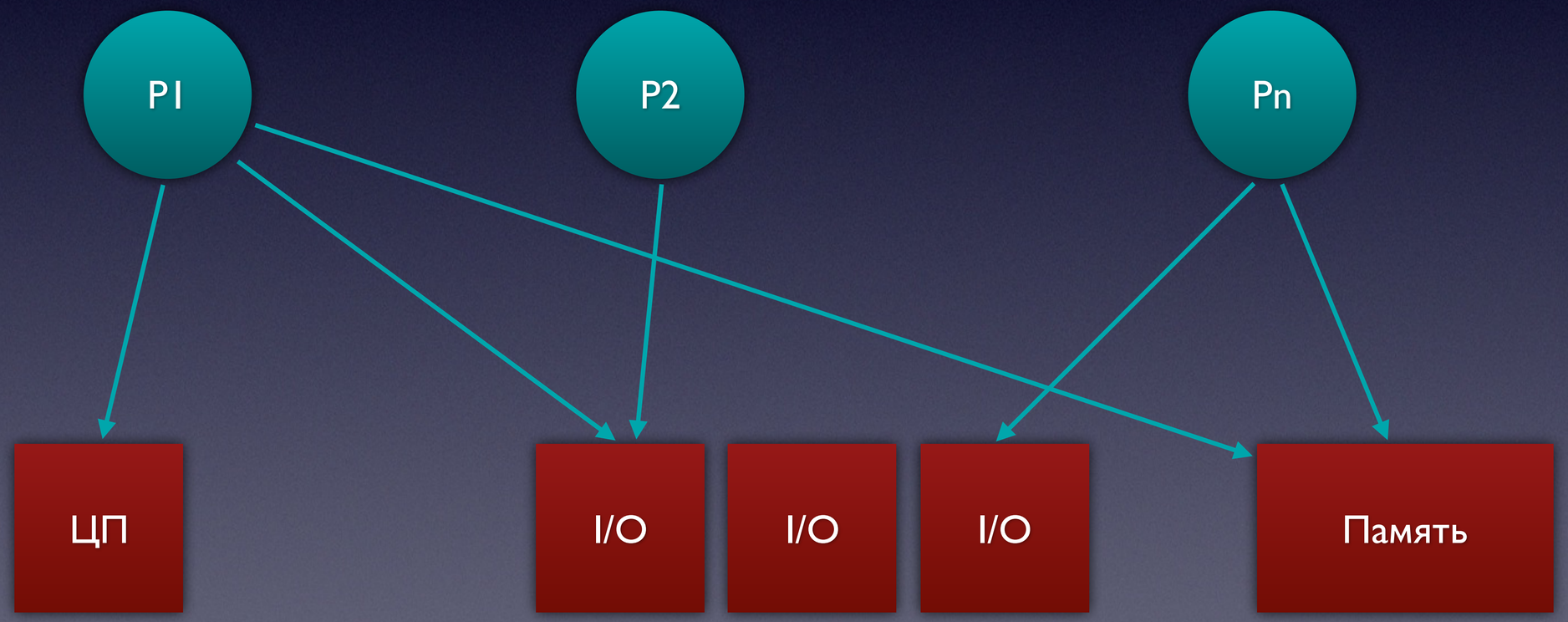
Два приостановленных состояния



Возможные причины остановки

- Swapping
- Решение ОС
- Решение пользователя
- Расписание
- Решение родительского процесса

Процессы и ресурсы



Контрольные структуры ОС

- ОС должна иметь информацию о текущем статусе процессов и ресурсов
- Для каждой сущности, управляемой ОС, создается таблица

- Память
- Устройства ввода/вывода
- Файлы
- Процессы

Таблицы памяти

- Отвечают за основную и вторичную память
  - Предоставление основной памяти процессам
  - Предоставление вторичной памяти процессам
  - Атрибуты защиты для доступа к общей памяти
  - Информация, необходимая для работы виртуальной памяти

Таблицы ввода/вывода

- Устройства ввода/вывода и каналы

- Системе необходимо знать
  - Доступно ли устройство
  - Статус операции ввода/вывода
  - Адрес в памяти, который используется для ввода или вывода

Файловые таблицы

- Информация о
  - Существовании файлов
  - Расположение во вторичной памяти
  - Текущий статус

Таблицы процессов

- Для эффективного управления системе необходимо знать
  - Идентификатор процесса
  - Текущее состояние процесса
  - Расположение в памяти
  - и т.д.

Режимы работы

- Пользовательский режим
- Системный режим

Создание процесса

- Присвоение уникального идентификатора
- Предоставление памяти процессу
- Инициализация контрольного блока
- Создание или изменение существующих структур

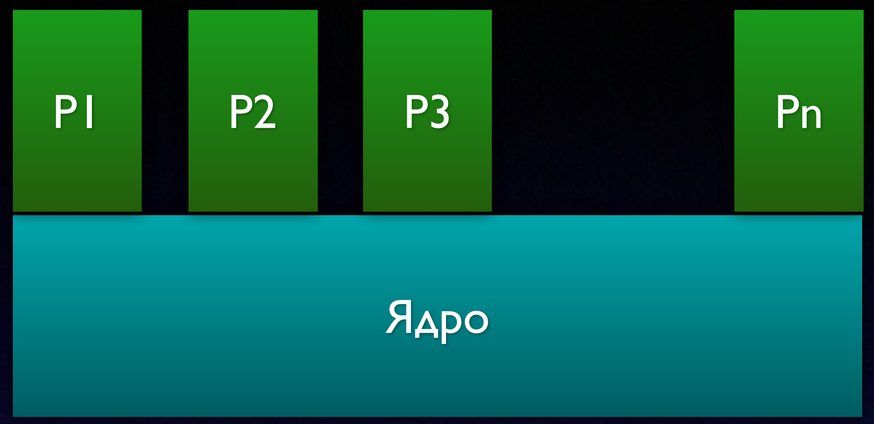
Переключение процесса

- Какой процесс стимулировал переключение?
- Что ОС должна делать со всеми структурами при переключении?

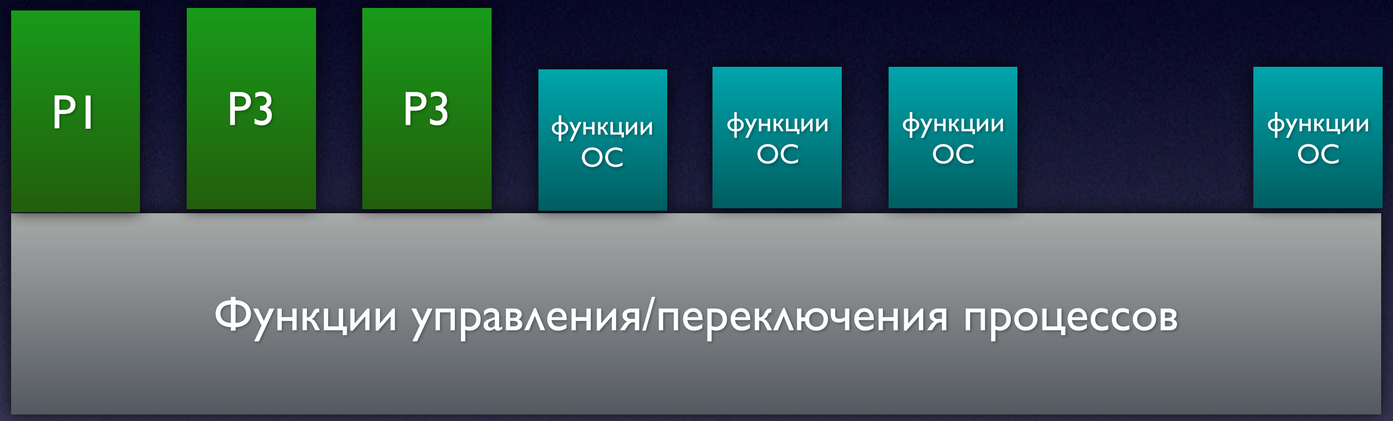
Переключение процесса

1. Сохранить состояние процесса (регистры)
2. Обновить контрольный блок процесса который запущен в данный момент
3. Перенести контрольный блок процесса в соответствующую очередь – ready; blocked; ready/suspend
4. Выбрать другой процесс для выполнения.
5. Обновить контрольный блок процесса, на который необходимо переключиться
6. Обновить структуры данных управления памятью
7. Восстановить состояние процесса

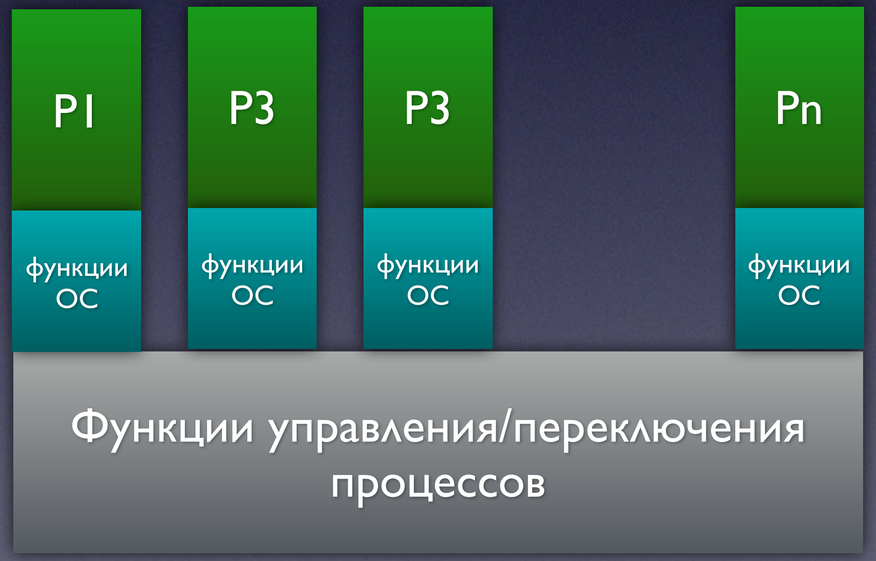
ОС – это процесс?



Отдельное ядро



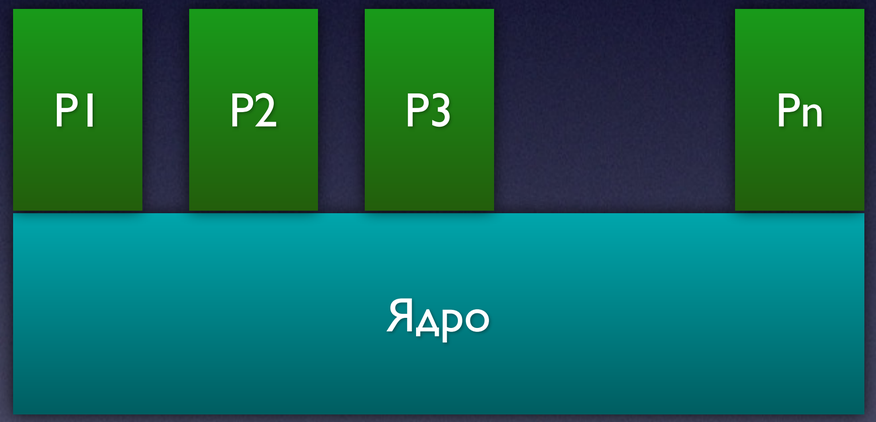
Функции ОС как отдельные процессы



Функции ОС в пользовательских процессах

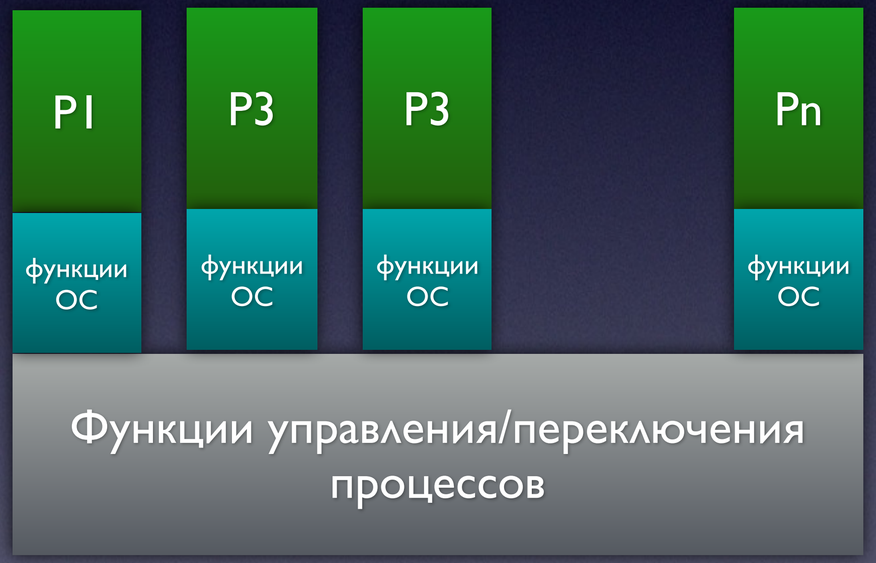
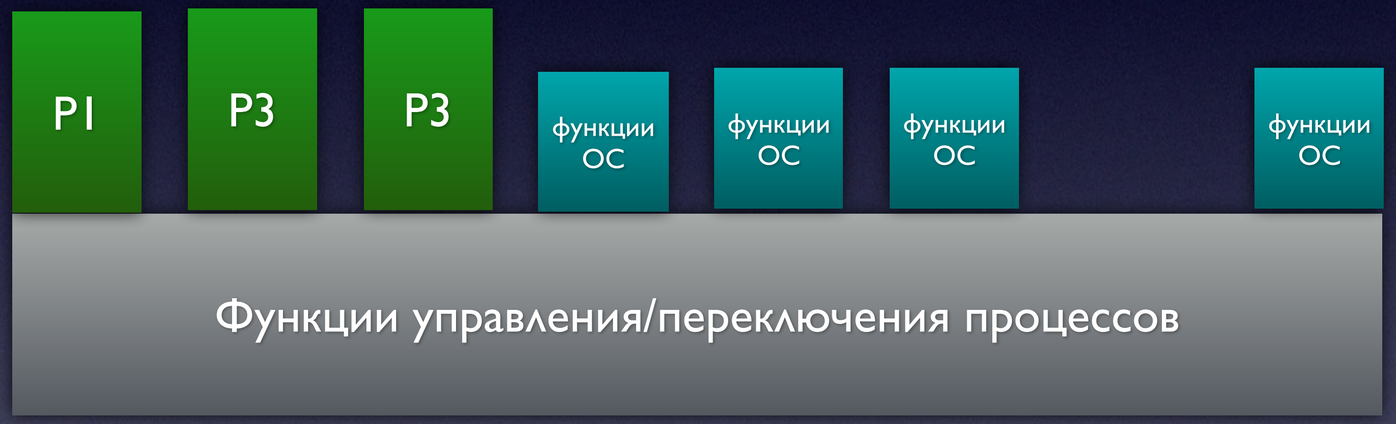
Отдельное ядро

- Ядро запущено вне процессов
- Ядро находится в отдельной области памяти
- Ядро функционирует в привилегированном режиме
- Понятие процесса относится только к пользовательским программам



Функции ОС в пользовательских процессах

- ОС предоставляет свои услуги “внутри” (в контексте) пользовательского процесса
- Не нужно переключаться для выполнения задач ОС



Функции ОС как отдельные процессы